



OPX-ST Family

(Operating with new firmware version 2.32)

Data-logger Functionality

Contents

0	Introduction.....	4
0.1	GENERAL	4
0.2	DATA-LOGGER FEATURES	4
1	Preparing to use the data-logger functionality.....	5
1.1	TRACKING.....	5
1.1.1	History function	5
2	NMEA messages – supported from integrated data-logger in the firmware version 2.32.....	11
2.1	NMEA MESSAGES SYNTAX.....	11
2.2	CONFIGURATION OF THE GPS LOGGING PARAMETERS	12
2.2.1	Set the NMEA messages that will be logged in the memory	12
2.2.2	Configure the history data	13
2.2.3	Get the history configuration	14
2.2.4	Enter a text to be written in the history	14
2.2.5	Download the history data.....	14
2.2.6	Clear all history records.....	15
2.3	NMEA MESSAGES SELECTED ON THE GPS DEVICE	15
2.3.1	Description of NMEA output messages	16
2.4	DESCRIPTION OF THE BINARY FORMAT	20
2.5	ABBREVIATIONS	20
2.6	THE FACTORY SETTINGS OF THE DATA-LOGGER FIRMWARE.....	22

Version history

Version number	Author	Changes
1.01	jssc	Initial release.

Cautions

Information furnished herein by OPANIEL is believed to be accurate and reliable. However, no responsibility is assumed for its use. Also the information contained herein is subject to change without notice.

Please, read carefully the safety precautions.

If you have any technical questions regarding this document or the product described in it, please contact your vendor.

General information about OPANIEL and its range of products is available at the following Internet address: <http://www.OPANIEL.com>

Trademarks

Some mentioned products are registered trademarks of their respective companies.

Copyright

This manual is copyrighted by OPANIEL with all rights reserved. No part of this user's guide may be produced in any form without the prior written permission of OPANIEL .

OPANIEL .

No patent liability is assumed with respect to the use of the information contained herein.

0 Introduction

0.1 General

The new firmware version 2.32, supplied by OPANIEL Technologies, consists of a data-logger application that can locally be configured. The data-logger functionality as an enhancement feature of the firmware 2.32 builds a fundamental component, which in combination with the excellent hardware performance makes the GPS units to be on the top of applications where the tracking systems are required. The concept of the firmware version 2.32 is based on a simple implementation for a wide range of applications such as vehicle tracking, transport systems, road-verifying systems etc.

This description is applied for all GPS units and GPS devices; the ordering name ends with DL character (e.g. OPX-ST-DL, etc.). Please refer to the hardware manual or user guide of the purchased (used) GPS unit or GPS device for more details respectively.

0.2 Data-logger features

With the data-logger functionality is possible to:

- ✓ record GPS position data in the on-board FLASH memory (about 180.000 records possible). Throughout the documentation is used as "History". The storage capacity depends on the OPX-ST integrated FLASH memory;
 - the standard version of the OPX-ST-DL GPS receiver consists of 8 Mbits memory (about 70.000 records possible)
 - the OPX-ST-DL GPS receiver (hardware option) with additional storage capacity (16 Mbits memory with upto 180.000 records possible) can be delivered upon request.
- ✓ prevent the history from storing unnecessary data (filter settings).
- ✓ download all or a part of the history stored data.
- ✓ filter the tracking mode using combinations between time, distance and velocity.
- ✓ read the configuration
- ✓ clear all the history records and many others

However, all features mentioned above do not require any hardware extensions on the GPS unit.

Please **note** that, when the target device operating with the data-logger firmware is initially turned on and either all minimum default settings or just one maximum default settings are/is exceeded, the GPS receiver starts automatically storing the current GPS position data into the on-board FLASH memory section using the factory default settings added in chapter 2.6.

Using the SiRFdemo Software, the factory setting (see chapter 2.6) can be changed and adapted to the user requirements. This configuration tool is available on the OPANIEL homepage: www.opaniel.com > [Downloads](#) and select the [SiRF](#) item from the selection box, then download the "SiRFdemo3.61.zip".

1 Preparing to use the data-logger functionality

1.1 Tracking

The GPS satellites are continuously transmitting a radio message containing information, including when the data was sent. When the GPS receiver fitted in the vehicle is powered on, through the connected GPS antenna, it receives continuously the radio messages sent from satellites containing GPS information the sent data. The received information from at least 4 satellites will be computed and carried out the necessary calculations to determine its current position. This data solution can be stored in the on-board Flash memory for further evaluation.

1.1.1 History function

The History memory enables the receiver to store position and time in the on-board FLASH memory. When receiving valid GPS protocols, GPS unit operating with firmware version 2.32 (with integrated data-logger functionality) is capable of saving up to 180.000 position records (hardware option, with additional storage capacity) in its history memory. When the memory space has been used up, the oldest records will automatically be deleted to make space for new incoming data (FIFO principal). Image 1 shows a logical flow chart, for better understanding the functionality of data logger in the GPS receiver, which stores the history data.

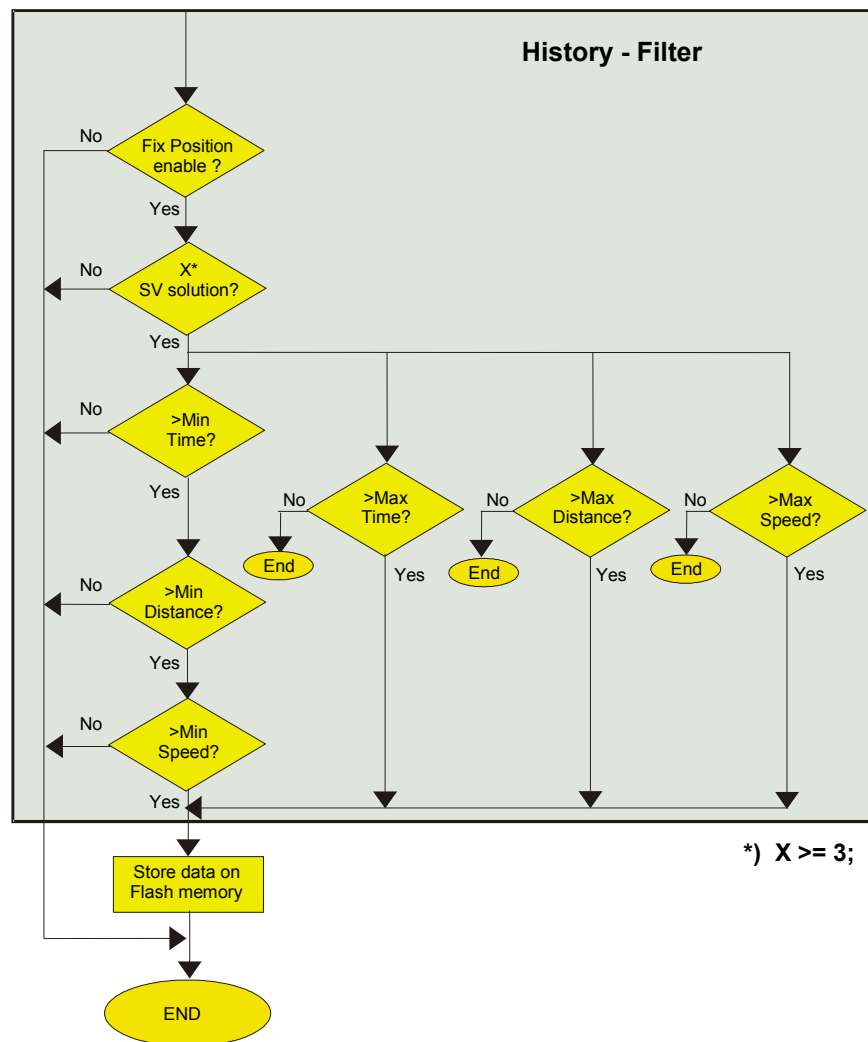


Image 1: History Filter

Taking the illustrated logic into consideration, several varieties of saving history data are also possible.

1.1.1.1 Filter Settings

By configuring supplementary filters, the history function offers the possibility to reduce the number of stored data records. These filters prevent the history from storing unnecessary data, e.g. if a vehicle is not moving or if a vehicle is moving within limited parameters.

To set supplementary filters, different one from the factory settings (see chapter 2.6), a configuration command can locally be sent to the GPS unit (by connecting the device to a PC) see also chapters 2.2.1 and 2.2.2.

1.1.1.2 Configuration examples

The following examples will explain the function of the filters for time, distance and speed.

The respective minimum values for time, speed and distance have an AND-conjunction, that means if minimum one value does not apply, the other values would be ignored and no history data would be saved.

The maximum values have an OR-conjunction. If one of the maximum parameters has been met, the GPS protocols will be saved in the history. In accordance with the above flow chart, either all minimum parameters or just one maximum parameter have to be exceeded, and the NMEA protocols are stored in the history.

Parameter explanation:

The parameters used in the examples below such as the *FixMin Time*, *FixMax Time*, *Dist.Min*, *Dist.Max*, *SpeedMin* and *Speed Max* correspond to the *<fmin>*, *<fmax>*, *<dmin>*, *<dmax>*, *<smmin>* and *<smmax>* used in the NMEA messages (see chapter 2.2) respectively.

	Minimum (lowest value)	Maximum (highest value)
Time filter	FixMinTime (Seconds)	FixMaxTime (Seconds)
	AND	OR
Distance filter	Dist.Min (Meters)	Dist.Max (Meters)
	AND	OR
Speed filter	SpeedMin (km/h)	SpeedMax (km/h)
Combination of both		OR
		Storing of protocol

Table 1: Conditions for saving data in the history

	FixMin Time [s]	FixMax Time [s]	Dist. Min[m]	Dist. Max [m]	SpeedMin [km/h]	Speed Max [km/h]
Logs every 10 sec.	0	10	0	0	0	0
Logs every 100 m	0	0	0	100	0	0
Logs every second at speed ≥ 40 km/h	0	0	0	0	0	40

Table 2: Programming example

1.1.1.3 Examples of history configuration

The GPS unit can be configured to store history depending on various conditions. When history logging is enabled and the minimum number of satellites is visible, history records will be stored depending on the time, distance and speed conditions. The history records would be stored in the NMEA format as specified in Section 2.3.1. Below are some configuration examples, which can be sent to the connected GPS unit.

Example 1

- ✓ Record every 50 m if the vehicle has a speed less then 25 km/h
- ✓ Record every 2 sec. if the vehicle is moving at a speed of between 25 and 50 km/h.
- ✓ Record every second if the vehicle is driving faster than 50 km/h

	FixMin Time [s]	FixMax Time [s]	Dist.Min [m]	Dist.Max [m]	SpeedMin [km/h]	Speed Max [km/h]
Example 1, settings NMEA message	2	0	0	50	25	50
\$PSRF109,11,4,2,0,0,50,25,50*2E						

Example 2

- ✓ Record every 50m if the vehicle is driving faster than 25 km/h

	FixMin Time [s]	FixMax Time [s]	Dist.Min [m]	Dist.Max [m]	SpeedMin [km/h]	Speed Max [km/h]
Example 2, settings NMEA message	0	0	50	0	25	0
\$PSRF109,11,4,0,0,50,0,25,0*19						

Example 3

- ✓ Record every 2 sec. if the vehicle is moving at a speed of between 25 and 50 km/h

	FixMin Time [s]	FixMax Time [s]	Dist.Min [m]	Dist.Max [m]	SpeedMin [km/h]	Speed Max [km/h]
Example 3, settings NMEA message	2	0	0	0	25	50
\$PSRF109,11,4,2,0,0,0,25,50*1B						

Example 4

- ✓ Record every second if the vehicle is driving faster than 50 km/h

	FixMin Time [s]	FixMax Time [s]	Dist.Min [m]	Dist.Max [m]	SpeedMin [km/h]	Speed Max [km/h]
Example 4, settings NMEA message	0	0	0	0	0	50
\$PSRF109,11,4,0,0,0,0,0,50*2E						

Other possible configuration

The settings in the table below indicate that a history record can be stored if

- ✓ since the last history record was stored, at least 300 seconds (5 minutes) have passed and the vehicle has traveled for at least 1,000 meters (1 kilometer), or
- ✓ 3600 seconds (1 hour) have passed since the last history record was stored, or
- ✓ the speed is higher than 100 kph.

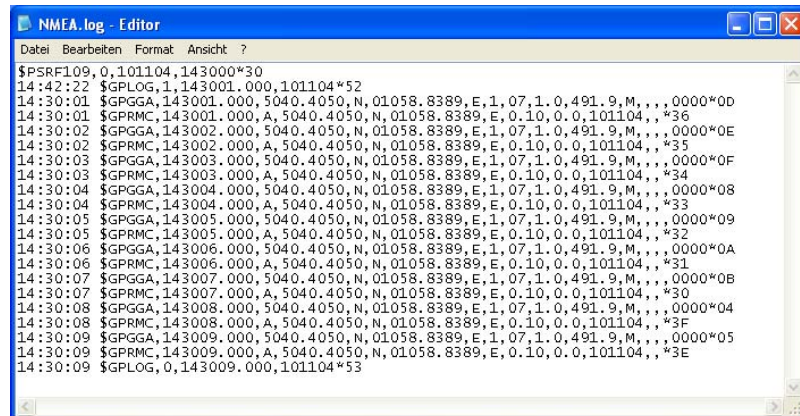
	FixMin Time [s]	FixMax Time [s]	Dist.Min [m]	Dist.Max [m]	SpeedMin [km/h]	Speed Max [km/h]
Example 5, settings NMEA message	300	3600	1000	0	0	100
\$PSRF109,11,4,300,3600,1000,0,0,0,100*01						

1.1.1.4 How to download the history records

The stored GPS history data can be downloaded from a direct connected GPS unit using the NMEA message described in chapter 2.2.2. Before the NMEA message is sent to the GPS unit, the used terminal program can be configured so that the incoming data can be captured into a *.log extended file format.

After the GPS history data are successfully downloaded and captured into a file on PC, the evaluation of the stored records can be performed using mapping software (Map&Guide or Microsoft MapPoint, etc) and a suitable user developed program (the user developed program reads the GPS data position from the stored *.log file and sent them to a COM port where the Map software is connected). To do this, open the map software and select the COM port where the data will be read. Start the user developed program and send the read data from the *.log file to the COM port where the Map software is connected. The route by means of the GPS data position from the selected file is being displayed in the map. Please, note that, if the stored records are in the binary format, they could not be accepted by the mapping software. A description about this format is attached in chapter 2.4.

1.1.1.5 NMEA and binary history data

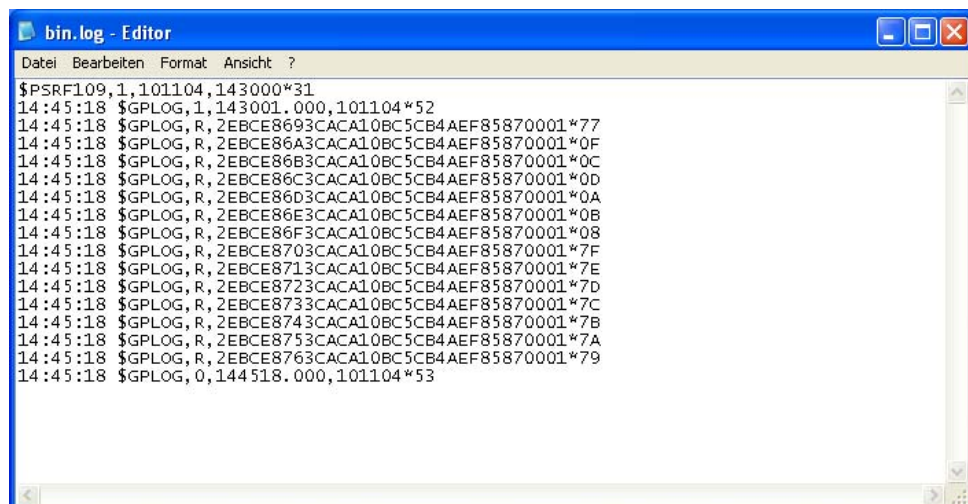


```

NMEA.log - Editor
Datei Bearbeiten Format Ansicht ?
$PSRF109,0,101104,143000*30
14:42:22 $GPLOG,1,143001.000,101104*52
14:30:01 $GPGGA,143001.000,5040.4050,N,01058.8389,E,1,07,1.0,491.9,M,,0000*0D
14:30:01 $GPRMC,143001.000,A,5040.4050,N,01058.8389,E,0.10,0.0,101104,,*36
14:30:02 $GPGGA,143002.000,5040.4050,N,01058.8389,E,1,07,1.0,491.9,M,,0000*0E
14:30:02 $GPRMC,143002.000,A,5040.4050,N,01058.8389,E,0.10,0.0,101104,,*35
14:30:03 $GPGGA,143003.000,5040.4050,N,01058.8389,E,1,07,1.0,491.9,M,,0000*0F
14:30:03 $GPRMC,143003.000,A,5040.4050,N,01058.8389,E,0.10,0.0,101104,,*34
14:30:04 $GPGGA,143004.000,5040.4050,N,01058.8389,E,1,07,1.0,491.9,M,,0000*08
14:30:04 $GPRMC,143004.000,A,5040.4050,N,01058.8389,E,0.10,0.0,101104,,*33
14:30:05 $GPGGA,143005.000,5040.4050,N,01058.8389,E,1,07,1.0,491.9,M,,0000*09
14:30:05 $GPRMC,143005.000,A,5040.4050,N,01058.8389,E,0.10,0.0,101104,,*32
14:30:06 $GPGGA,143006.000,5040.4050,N,01058.8389,E,1,07,1.0,491.9,M,,0000*0A
14:30:06 $GPRMC,143006.000,A,5040.4050,N,01058.8389,E,0.10,0.0,101104,,*31
14:30:07 $GPGGA,143007.000,5040.4050,N,01058.8389,E,1,07,1.0,491.9,M,,0000*0B
14:30:07 $GPRMC,143007.000,A,5040.4050,N,01058.8389,E,0.10,0.0,101104,,*30
14:30:08 $GPGGA,143008.000,5040.4050,N,01058.8389,E,1,07,1.0,491.9,M,,0000*04
14:30:08 $GPRMC,143008.000,A,5040.4050,N,01058.8389,E,0.10,0.0,101104,,*3F
14:30:09 $GPGGA,143009.000,5040.4050,N,01058.8389,E,1,07,1.0,491.9,M,,0000*05
14:30:09 $GPRMC,143009.000,A,5040.4050,N,01058.8389,E,0.10,0.0,101104,,*3E
14:30:09 $GPLOG,0,143009.000,101104*53

```

The stored position data in NMEA format begins with \$GPLOG,1,..... and ends with \$GPLOG,0,..... which indicate the start date/time (10 November 2004 14:30:01) and end date/time (10 November 2004 14:30:09). All records between the start and end records include the stored GPS position data. A description about the NMEA messages can be found in chapter 2.3. The "\$PSRF109,0,101104,143000*30" NMEA command containing the start date/time is the user sent message to download the flash memory data.



```

bin.log - Editor
Datei Bearbeiten Format Ansicht ?
$PSRF109,1,101104,143000*31
14:45:18 $GPLOG,1,143001.000,101104*52
14:45:18 $GPLOG,R,2EBCE8693CACA10BC5CB4AEF85870001*77
14:45:18 $GPLOG,R,2EBCE86A3CACA10BC5CB4AEF85870001*0F
14:45:18 $GPLOG,R,2EBCE86B3CACA10BC5CB4AEF85870001*0C
14:45:18 $GPLOG,R,2EBCE86C3CACA10BC5CB4AEF85870001*0D
14:45:18 $GPLOG,R,2EBCE86D3CACA10BC5CB4AEF85870001*0A
14:45:18 $GPLOG,R,2EBCE86E3CACA10BC5CB4AEF85870001*0B
14:45:18 $GPLOG,R,2EBCE86F3CACA10BC5CB4AEF85870001*08
14:45:18 $GPLOG,R,2EBCE8703CACA10BC5CB4AEF85870001*7F
14:45:18 $GPLOG,R,2EBCE8713CACA10BC5CB4AEF85870001*7E
14:45:18 $GPLOG,R,2EBCE8723CACA10BC5CB4AEF85870001*7D
14:45:18 $GPLOG,R,2EBCE8733CACA10BC5CB4AEF85870001*7C
14:45:18 $GPLOG,R,2EBCE8743CACA10BC5CB4AEF85870001*7B
14:45:18 $GPLOG,R,2EBCE8753CACA10BC5CB4AEF85870001*7A
14:45:18 $GPLOG,R,2EBCE8763CACA10BC5CB4AEF85870001*79
14:45:18 $GPLOG,0,144518.000,101104*53

```

The stored position data in binary format also begins with \$GPLOG,1,..... and ends with \$GPLOG,0,..... which indicate the start date/time (10 November 2004 14:30:01) and end date/time (10 November 2004 14:30:09). All record data between the start and end lines include the stored GPS data in the binary format. A description about this format is attached in chapter 2.4. This format offers a maximum data compression. The "\$PSRF109,1,101104,143000*31" NMEA command containing the start date/time is the user sent message to download the flash memory data.

2 NMEA messages – supported from integrated data-logger in the firmware version 2.32

The data is transmitted in the form of "sentences". The sentence begins with start "\$" character, next come the four letters "talker ID" and three letter "sentence ID", followed by a number of parameters separated by commas, and terminated by a calculated checksum, and a carriage return/line feed (see table below). These last two characters <CR><LF> are "control" characters and are normally not printed (for this reason they are customarily shown in brackets).

"\$"	"talker ID"	"sentence ID"	parameters	checksum	carriage return/line feed
\$	PSRF	109	<Data>	<*CKSUM>	<CR><LF>

The configuration parameters are temporarily stored in the SRAM. In order to reduce the start up time of the receiver and to keep the user-specified configuration it is preferable to connect an external back up battery on the target GPS unit (means, supply the RTC of GPS unit from a separate voltage e.g. battery), so that the RTC is running during the power interrupt. The backup battery power is required for retention of SRAM memory and maintaining the Real-Time-Clock. The validity of data and user-configuration stored in SRAM will be kept due to RTC keeps running and these data will be available on the next power up scenario. If no external backup battery is connected to the GPS unit, the validity of these data and user-performed configuration stored in SRAM are lose and after each power-up the device will be returned back to the factory settings (see chapter 2.6). Maybe new configuration is required to be performed after each power-up.

2.1 NMEA Messages syntax

The NMEA standard messages have the following format:

Command	Parameter	Checksum	End Sequence
\$PSRF109	<Data>	<*CKSUM>	<CR><LF>

The \$PSRF message header and the message identifier "109" creates a NMEA command.

The parameter <Data> can contain different valid parameters, which are supported by the internal firmware.

The checksum <*CKSUM> consists of a "*" character followed by two hex values.

In order to calculate the Checksum, use your own application. Below a small source code written in Visual Basic:

```
{*****}
Public Sub CheckSum(field As String)
If field = "" then CS = "*"
CS = 0
For i = 1 to Len(field)
CS = CS Xor Asc(Mid$(field, i, 1))
Next
CS = Hex(CS)
If Len(CS) = 1 then
CS = "0" & CS
CS = "*" & CS
END SUB
{*****}
```

Therefore, the string over which the checksum has to be calculated is:

field = PSRF109,<data>
excluding "\$" character.

All fields in all proprietary NMEA messages are required, none are optional.

All parameters of NMEA messages are comma delimited ",".

2.2 Configuration of the GPS logging parameters

2.2.1 Set the NMEA messages that will be logged in the memory

Syntax	\$PSRF103,<msg>,00,<rate>,<Cksum_enable><*CkSum><CR><LF>
Examples	<pre>\$PSRF103,00,00,00,01*24 // Deactivate GPGGA \$PSRF103,00,00,01,01*25 // Activate GPGGA \$PSRF103,01,00,00,01*25 // Deactivate GPGLL \$PSRF103,01,00,01,01*24 // Activate GPGLL \$PSRF103,02,00,00,01*26 // Deactivate GPGSA \$PSRF103,02,00,01,01*27 // Activate GPGSA \$PSRF103,03,00,00,01*27 // Deactivate GPGSV \$PSRF103,03,00,01,01*26 // Activate GPGSV \$PSRF103,04,00,00,01*20 // Deactivate GPRMC \$PSRF103,04,00,01,01*21 // Activate GPRMC \$PSRF103,05,00,00,01*21 // Deactivate GPVTG \$PSRF103,05,00,01,01*20 // Activate GPVTG</pre>

Command Description

This command is used to control the output of supported NMEA messages GGA, GLL, GSA, GSV, RMC and VTG.

Parameter Description

<msg>

It specifies the NMEA message that will be configured. Following values are available:

- 00 corresponds to the [GPGGA](#) message
- 01 corresponds to the [GPGLL](#) message
- 02 corresponds to the [GPGSA](#) message
- 03 corresponds to the [GPGSV](#) message
- 04 corresponds to the [GPRMC](#) message
- 05 corresponds to the [GPVTG](#) message

00

predefined value (set the NMEA messages)

<rate>

It specifies the message period in seconds. Following values can be set:

- 00 disables the specified message
- 01 enables the specified message

<Cksum_enable>

It specifies the enabling/disabling of checksums. Checksums may also be enabled or disabled depending on the needs of the receiving program.

Notes

- If the field <rate> is specified to 00, the selected message <msg> will be disabled (switched OFF).
- The configuration of the NMEA message is saved in battery-backed memory for each entry when the message is accepted.

- These user-changes are temporarily stored and will be lost after each power down of the GPS unit and the configuration of the unit will be returned back to the factory settings, if no external backup battery is connected.
- It is necessary to note, that all NMEA messages supported by the data-logger firmware can be activated/deactivated by using this command, however, only the [GPGGA](#), [GPGLL](#) and [GPRMC](#) messages can be stored in the history section of the on-board FLASH memory (data-logger functionality).

2.2.2 Configure the history data

Syntax	\$PSRF109,11,<sat>,<fmin>,<fmax>,<dmin>,<dmax>,<smin>,<smax><*CkSum><CR><LF>
--------	--

Configuring triggers specifies to the GPS unit under what circumstances a position should be logged into memory. This NMEA message allows you to specify/modify the parameters of the history data. The history logging depends on the in view satellites and the minimum or maximum time, distance and speed. If all minimum values or just one maximum value is exceeded, then the GPS unit begins storing the specified NMEA protocols in the history.

New configurations take effect immediately, so they become permanently enabled until changed.

<sat>

It determines the minimum number of in view satellites able to start the history logging.

<fmin>

It specifies the minimum time. The unit is second [s]

<fmax>

It specifies the maximum time. The unit is second [s]

<dmin>

It specifies the minimum distance. The unit is meter [m]

<dmax>

It specifies the maximum distance. The unit is meter [m]

<smin>

It specifies the minimum speed. The unit is kilometer per hour [km/h]

<smax>

It specifies the maximum speed. The unit is kilometer per hour [km/h]

*<*CkSum>*

The checksum has to be calculated.

How this message could be sent:

Example	\$PSRF109,11,4,5,3600,0,0,0,0*2B
---------	----------------------------------

Notes

- To understand and to set properly the *<tmin>*, *<tmax>*, *<dmin>* and *<dmax>*, *<smin>*, *<smax>* parameters, please, refer to the chapter 1.1.1. In that chapter you will find detailed information.
- If all *<tmin>*, *<dmin>* and *<smin>* limits are exceeded, a history record will be stored. Any null value is ignored.
- If any of *<fmax>*, *<dmax>*, *<smax>* limits is exceeded, a history record will be stored. Any null value is ignored.
- The user-configuration will temporarily be stored in the SRAM memory and it is valid as long as the RTC is running.
- If all parameters are set to 0, the history logging is disabled.

2.2.3 Get the history configuration

Syntax	\$PSRF109,12<*CkSum><CR><LF>
--------	------------------------------

This command is a helper method that reads out the history configuration from the target unit and fills them in the respond message.

How this message could be sent:

Example	\$PSRF109,12*00
---------	-----------------

Responds	\$GPLOG,P,<sat>,<fmin>,<fmax>,<dmin>,<dmax>,<smin>,<smax> <*CkSum><CR><LF>
-----------------	---

Example	\$GPLOG,P,11,4,5,3600,0,0,0,0*28
----------------	----------------------------------

Notes

- This command reads out the history configuration such as the <sat>,<fmin>,<fmax>,<dmin>,<dmax>,<smin> and <smax> from the \$PSRF109,11 configuration message of the target unit and fills them in the respond message respectively.
- The P character indicates that the message contains the history configuration parameters.

2.2.4 Enter a text to be written in the history

Syntax	\$PSRF109,<type>,<text><*CkSum><CR><LF>
--------	---

This command allows you to enter the user-specified text into the history memory.

<type>

It specifies the type of entered <text> format. The type of text can be configured in the following formats:

20 enables the <text> field to be entered in the **TEXT** format

21 enables the <text> field to be entered in the **binary** format.

<text>

It specifies the user-defined text, which can be written in the history of device for user requirements. Upto 508 bytes are available.

<*CkSum>

The checksum has to be calculated.

How this message could be sent:

Example	\$PSRF109,20,the_text_which_will_be_entered_in_the_history*27 \$PSRF109,21,11AA22BB33CC44DD55EE66FF*26
----------------	---

2.2.5 Download the history data

Syntax	\$PSRF109,<r_data>,<s_date>,<s_time>,<e_date>,<e_time> <*CkSum><CR><LF>
--------	---

Command Description

Download history from GPS unit using specified parameters. The start date/time and end date/time options allow downloading only of data within a certain date range. The data into the on-board memory is stored according to UTC Time (Universal

Time Coordinated). Therefore, the parameters of this command are also based on the UTC time.

Command Description

<r_data>

It specifies the download history format. The default setting for the download date format is set to 0 (NMEA format). The GPS data can be configured in the following data formats:

- 0 requests the history data in the **NMEA** format
- 1 requests the history data in the **binary** format

<s_date>

Specifies the start date. Its format is DDMMYY

<s_time>

Specifies the start time. Its format is HHMMSS

<e_date>

Specifies the end date. Its format is DDMMYY

<e_time>

Specifies the end time. Its format is HHMMSS

<*CkSum>

Checksum has to be calculated.

How this message could be sent:

Examples	<pre>\$PSRF109,0,280104,155811,300104,150000*37 \$PSRF109,0,280104,155811*35 \$PSRF109,0,0,0*33 \$PSRF109,1,280104,155811,300104,150000*36 \$PSRF109,1,0,0*32</pre>
-----------------	---

Notes

- The message parameters **<s_date>** and **<s_time>** start date/time are required, none are optional, whereas the **<e_date>** and **<e_time>** end date/time are optional.
- All history data will be downloaded, if the start date/time and the end date/time option fields are set to 0.
- To download the history data, please consider the UTC Time, otherwise you will download the stored history data in the incorrect time.

2.2.6 Clear all history records

Syntax	\$PSRF109,10<*CkSum><CR><LF>
--------	------------------------------

Command Description

This NMEA message allows you to clear all history records from the GPS unit memory. The value "10" is predefined one.

How this message could be sent:

Examples	\$PSRF109,10*02
-----------------	-----------------

2.3 NMEA messages selected on the GPS device

The GPS receiver transmits NMEA sentences every second. The identifiers for the NMEA messages transmitted by the GPS receiver are listed below.

- ✓ **GPGGA** GPS Fix Data
- ✓ **GPRMC** Recommended Minimum Specific GPS Data

- ✓ **GPGSV** GPS Satellites in View
- ✓ **GPGSA** GPS DOP and Active Satellites
- ✓ **GPGLL** Geographic Position in Latitude/Longitude.
- ✓ **GPVTG** Course Over Ground and Ground Speed

Only the **GPGLL**, **GPRMC** and **GPGLL** NMEA messages can be stored into the on-board FLASH memory.

The next subsection provides a full description and definition of the listed NMEA messages.

2.3.1 Description of NMEA output messages

The following table is intended as a quick reference to explain the formats used in the tables below.

Format	Description
hhmmss.ss	Time: hh hours, mm minutes, ss.ss seconds.
ddmmyy	Date: day dd, month mm, year yy.
ddmm.mmmm	Latitude: dd degrees, mm.mmmm minutes.
dddmm.mmmm	Longitude: ddd degrees, mm.mmmm minutes.
dd.dddddd	Latitude/longitude: dd.dddddd degrees.
dd'mm'ss"	Latitude/longitude: dd degrees, mm minutes, ss seconds
x	Integer.
xx	Integer having exactly two digits (using leading zeros).
x.x	Number including fraction.
hh	Two-digit hexadecimal number (using uppercase A–F).
bbbbbbbb	Eight-digit binary number.
a	ASCII text.
"a"	ASCII text in quotation marks.
<CR><LF>	Carriage return and line feed.

Table 3: A quick reference of the GPS messages format.

The \$GPGGA message includes time, position, GPS quality and number of satellites in use.

Example:

\$GPGGA,105614.387,5040.4010,N,01058.8426,E,1,10,0.8,491.4,M,47.7,M,0.0,0000*7C

Format	Example	Description
\$GPGGA	\$GPGGA	Start of sentence
hhmmss.ss	105614.387	UTC time
ddmm.mmmm	5040.4010	Latitude
a	N	Latitude direction (N/S)
dddmm.mmmm	01058.8426	Longitude
a	E	Longitude direction (W/E)
x	1	GPS fix quality: 0: invalid 1: GPS fix 2: DGPS fix
xx	10	Number of satellites in use
x.x	0.8	Horizontal dilution of precision (relative accuracy of horizontal position)
x.x	491.4	Altitude above mean sea level (geoid)
M	M	Altitude units (meters)

x.x	47.7	Height of geoid above earth ellipsoid
M	M	Geoid height units (meters)
x	0.0	Time since last DGPS update (seconds)
xxxx	0000	DGPS reference station ID
*hh	*3F	Checksum
<CR><LF>		End of message termination

Table 4: The GPGBA message data format.

The \$GPRMC message includes time, date, position, course and speed data.

Example:

\$GPRMC,133725.569,A,5040.4365,N,01058.5650,E,0.05,302.98,251004,,*00

Format	Example	Description
\$GPRMC	\$GPRMC	Start of sentence
hhmmss.ss	133725.569	UTC time
a	A	Position validity (A: valid, V: invalid)
ddmm.mmmm	5040.4365	Latitude
a	N	Latitude direction (N/S)
dddmm.mmmm	01058.5650	Longitude
a	E	Longitude direction (W/E)
x.x	0.05	Speed (knots)
x.x	302.98	Heading (degrees)
ddmmyy	251004	Date
x.x		Magnetic variation (degrees)
a		Magnetic variation direction (W/E)
*hh	*00	Checksum
<CR><LF>		End of message termination

Table 5: The GPRMC message data format.

The \$GPGLL message includes the latitude, longitude, UTC time of position fix and status.

Example:

\$GPGLL,5040.4025,N,01058.8342,E,113704.665,A*32

Format	Example	Description
\$GPGLL	\$GPGLL	Start of sentence
ddmm.mmmm	5040.4025	Latitude
a	N	Latitude direction (N/S)
dddmm.mmmm	01058.8342	Longitude
a	E	Longitude direction (W/E)
hhmmss.sss	113704.665	UTC Position
a	A	Position validity (A: valid, V: invalid)
*hh	*32	Checksum
<CR><LF>		End of message termination

Table 6: The GPGLL message data format.

The \$GPGSV includes the number of satellites in view satellite ID numbers and their elevation, azimuth and signal-to-noise ratio.

Example:

\$GPGSV,3,1,09,06,73,068,39,25,44,247,39,21,41,177,41,30,33,126,40*78

Format	Example	Description
\$GPGSV	\$GPGSV	Start of sentence
x	3	Number of messages (1 to 3)
x	1	Message number (1 to 3)
xx	09	Number of satellites in view (1 to 12)
xx	06	Channel 1: Satellite ID (1 to 32)
xx	73	Channel 1: Satellite elevation (degrees) (00 to 90), may be null
xxx	068	Channel 1: Satellite azimuth (degrees) (000 to 359), may be null
xx	39	Satellite signal to noise ratio in dBHz (00 to 99), may be null
...
xx	30	Channel 4: Satellite ID (1 to 32)
	33	Channel 4: Satellite elevation (degrees) max. 90, may be null
	126	Channel 4: Satellite azimuth (degrees) (true, range 000 to 359), may be null
	40	Satellite signal to noise ratio in dBHz (00 to 99), may be null
*hh	*74	Checksum
<CR><LF>		End of message termination

Table 7: The GPGSV message data format.

The \$GPGSA message includes the list of satellites being used.

Example:

\$GPGSA,A,3,06,25,21,30,17,16,10,,,,,2.5,1.0,2.3*34

Format	Example	Description
\$GPGSA	\$GPGSA	Start of sentence
a	A	Operating mode: M: Manual, operate in 3-D mode. A: Automatically choose 2-D or 3-D mode.
x	3	Fix mode: 1: Fix not available 2: 2-D fix 3: 3-D fix
xx	06	Channel 1: PRN numbers of satellites in use
xx	25	Channel 2: PRN numbers of satellites in use
xx
xx		Channel 12: PRN numbers of satellites in use
x.x	2.5	Position dilution of precision
x.x	1.0	Horizontal dilution of precision
x.x	2.3	Vertical dilution of precision
*hh	*3D	Checksum
<CR><LF>		End of message termination

Table 8: The GPGSA message data format.

The \$GPGLL message includes the latitude, longitude, UTC time of position fix and status.

Format	Example	Description
\$GPGLL	\$GPGLL	Start of sentence
ddmm.mmmm	5040.4025	Latitude
a	N	Latitude direction (N/S)
dddmm.mmmm	01058.8342	Longitude
a	E	Longitude direction (W/E)
hhmmss.sss	113704.665	UTC Position
a	A	Position validity (A: valid, V: invalid)
*hh	*32	Checksum
<CR><LF>		End of message termination

Example:

\$GPGLL,5040.4025,N,01058.8342,E,113704.665,A*32

Table 9: The GPGLL message data format.

The \$GPVTG message reports the course over ground and ground speed information.

Example:

\$GPVTG,109.53,T,,M,0.15,N,0.3,K*69

Format	Example	Description
\$GPGSA	\$GPGSA	Start of sentence
xxx.xx	109.53	Course in degree
a	T	True
xxx.xx		Course in degree
a	M	Geodetic WGS84 directions
xxx.xx	0.15	Speed in knots
a	N	knots
xxx.xx	0.3	Speed in km/hr
a	K	Kilometers per hour
*hh	*69	Checksum
<CR><LF>		End of message termination

Table 10: The GPVTG message data format.

2.4 Description of the binary format

The binary format stored in the memory of GPS unit has the following structure.

Name	Indicator	Format	Checksum	End Sequence
\$GPLOG	R/T or X	<binary_data>	<*CKSUM>	<CR><LF>
\$GPLOG,R,2EA8BB7D3CACA20BC5C94AEF7E840002*0D<CR><LF>				

Indicator can be:

- R Indicates that the stored record contains the GPS position data in the binary format.
- T Indicates that the stored record contains the user-defined text in the NMEA format.
- X Indicates that the stored record contains the user-defined text in the binary format.

The following table is intended as a quick reference to explain the stored binary data.

Name	Bytes	Binary (hex)		Units	ASCII (Dec)	
		Scale	Example		Scale	Example
\$GPLOG	-	-	-	-	-	-
R	-	-	-	-	-	-
GPS time (since 01/06/1980)	4		2EA8BB7D	seconds		782809981
X-position	3		3CACA2	m		3976354
Y-position	3		0BC5C9	m		771529
Z-position	3		4AEF7E	m		4910974
Fix indicator and SVs in Fix *	1		84 (=80 + 4)			132 (=128 + 4) (see next table)
Velocity *0D	2	*20	0002	m/s	V÷20	0.1
	-	-	-	-	-	-

* For validated and invalidated solution definition, see table below.

Fix indicator and SVs in Fix		Description
Hex	ASCII	
0 x 00(+SV)	00(+SV)	Solution is not validate Solution is validate
0 x 80(+SV)	128(+SV)	
SVs in Fix		No navigation solution 1 Satellite solution 2 Satellite solution 3 Satellite solution > 3 Satellite solution
0 x 00	0	
0 x 01	1	
0 x 02	2	
0 x 03	3	
0 x 04 to 0 x 0C	4 to 12	
Example		
0 x 84	132	Solution is validate with 4 satellites in view

2.5 Abbreviations

Altitude The distance between the current position and the nearest point on WGS-84 (World Geodetic System 1984) reference ellipsoid. Altitude is usually expressed in meter an is positive outside the ellipsoid.

Current position	The GPS is based on satellite ranging - calculating the distances between the receiver and the position of 3 or more satellites (4 or more if elevation is desired) and then applying some good old mathematics. Assuming the positions of the satellites are known, the location of the receiver can be calculated by determining the distance from each of the satellites to the receiver. GPS takes these 3 or more known references and measured distances and "triangulates" an additional position.
DGPS	Differential GPS
FIFO	First In First Out, system in which the first item stored is the first item retrieved.
FLASH memory	A particular type of memory that is permanent and the data written on it will be not lost when the device is turned off. It can be updated and upgreated by special program.
GPS	The Global Positioning System (GPS) is a location system based on a constellation of about 24 satellites orbiting the earth at altitudes of approximately 11,000 miles. GPS satellites are orbited high enough to avoid the problems associated with land based systems, yet can provide accurate positioning 24 hours a day, anywhere in the world.
GPS Time	The number of seconds since Saturday/Sunday Midnight UTC, with time zero begging this midnight. Used with GPS Week Number to determine a specific point in GPS Time.
History	A limited part (section) on the on-board FLASH memory that serves to save the user-selected position records (the current data position received from the GPS satellites) and those data can be available for evaluation in a later time.
Latitude	Halfway between the poles lies equator. Latitude is the angular measurement of a place expressed in degrees north or south of the equator. Latitude runs from 0° at the equator to 90° north or 90° south at the poles. When not prefixed with letters E or W, it is assumed positive north of Equator and negative south of Equator. Lines of latitude run in an east-west direction. They are called parallels.
Locally	The GPS unit is directly connected to the host device (PC/Laptop etc.) through the serial interface and via supported commands the target device can be configured (e.g. the default settings can be changed and if required, the new configuration can be performed).
Longitude	Lines of longitude, called meridians, run in a north-south direction from pole to pole. Longitude is the angular measurement of a place east or west of the prime meridian. This meridian is also known as the Greenwich Meridian, because it runs through the original site of the Royal Observatory, which was located at Greenwich, just outside London, England. Longitude runs from 0° at the prime Meridian to 180° east or west, halfway around the globe. When not prefixed with letters E or W, it is assumed positive east of Greenwich and negative west of Greenwich. The International Date Line follows the 180° meridian, making a few jogs to avoid cutting through land areas.
NMEA	National Maritime Electronics Association

RTC	Real Time Clock continues running when the device is turned on or supplied by an external battery when the device is turned off.
SRAM	Static Random Access Memory, a type of memory that temporarily keeps its information that processor use to execute instructions. Should the device be switched off and no external backup battery connected, it lose the contents of SRAM forever.
UTC	Universal Time Coordinated
X, Y, Z positions	Coordinates of user's position in ECEF (meters). The Earth-centered Earth-Fixed (ECEF) is a Cartesian coordinate system with its origin located at the center of the Earth. The coordinate system used by GPS to describe 3-D location. For WGS-84 reference ellipsoid. ECEF coordinated have the Z-axis aligned with the Earth's spin axis, The X-axis through the insertion of the Prime meridian and the Equator and the Y-axis is rotated 90 degrees East of the X-axis about the Z-axis.

2.6 The factory settings of the data-logger firmware

	Sat	FixMin Time [s]	FixMax Time [s]	Dist.Min [m]	Dist.Max [m]	SpeedMin [km/h]	Speed Max [km/h]
Parameter	<sat>	<fmin>	<fmax>	<dmin>	<dmax>	<smin>	<smax>
Default settings	4	3	900	0	0	10	0

NOTE: Other factory settings can be implemented upon user-request.